

## ממ"ן 12 – פתרון

הקורס : 20471 (ארגון המחשב) 2023א

שאלה 1 (15%)

זיכרו : נתון שכתובת התווית Shark הוא 0x10203040

הפקודה	א. ייצוג בעשרוני עם רווח בין חלקי הפקודה השונים.
Shark:	0x10203040=270544960
lw \$12, -0x100(\$15)	opcode=0x23=35, rs=15, rt=12, immediate=-256
or \$15,\$22,\$11	opcode=0, rs=22, rt=11, rd=15, shamt=0, funct=0x25=37
beq \$12,\$27, Shark	opcode=0x4=4, rs=12, rt=27, immediate=-4
j Shark	opcode=0x2=2, addr=527376

הפקודה	ב. ייצוג בבינרי עם רווח בין חלקי הפקודה השונים, כל חלק פקודה עם הכותרת המתאימה (למשל rs=10100
Shark:	000100000010000000011000001000000
lw \$12, -0x100(\$15)	opcode=100011, rs=01111, rt=01100, immediate=1111 1111 0000 0000
or \$15,\$22,\$11	opcode=000000, rs=10110, rt=01011, rd=01111, shamt=00000, funct=100101
beq \$12,\$27, Shark	opcode=000100, rs=01100, rt=11011, immediate=1111 1111 1111 1100
j Shark	opcode=000010, addr=00 0000 1000 0000 1100 0001 0000

הפקודה	ג. ייצוג בבינרי ברצף 32 סיביות רצופות.
Shark:	0001 0000 0010 0000 0011 0000 0100 0000
lw \$12, -0x100(\$15)	1000 1101 1110 1100 1111 1111 0000 0000
or \$15,\$22,\$11	0000 0010 1100 1011 0111 1000 0010 0101
beq \$12,\$27, Shark	0001 0001 1001 1011 1111 1111 1111 1100
j Shark	0000 1000 0000 1000 0000 1100 0001 0000

הפקודה	ד. ייצוג ב-16 ספרות בבסיס 4 ברצף.
Shark:	0100020003004000
lw \$12, -0x100(\$15)	2031323033330000
or \$15,\$22,\$11	0002302313200211
beq \$12,\$27, Shark	0101212333333330
j Shark	0020002000300100

הפקודה	ה. ייצוג ב-11 ספרות בבסיס 8 ברצף.
Shark:	02010030100
lw \$12, -0x100(\$15)	21573177400
or \$15,\$22,\$11	00262674045
beq \$12,\$27, Shark	02146777774
j Shark	01002006020

הפקודה	ו. ייצוג ב-8 ספרות הקסאדצימלי ברצף.
Shark:	0x10203040
lw \$12, -0x100(\$15)	0x8decff00
or \$15,\$22,\$11	0x0cb7825
beq \$12,\$27, Shark	0x119bfff
j Shark	0x08080c10

## שאלה 2 (10%)

1. בסיום הפקודה בשורה (4):  $\text{and } \$t0, \$t4, \$t2$ , מה יהיה ערכו של אוגר  $t0$ ? הערך  $0xFF00FF00$  חישוב והסבר:

בפקודת הֹלִי ייכנס הערך המיידִי  $0xFF00FFFF$  לאוגר  $t4$ .  
בפקודת  $sra$  יתבצע שיפט אֶרִיתֻמִּי של ערך זה 16 מקומות ימינה, כלומר יתווספו 16 ספרות 1 בינאריות בשמאל המספר (ה־MSB של הערך הנוכחי ב־ $t4$  הוא 1).  
לכן, ייכנס ל־ $t2$  הערך  $0xFFFFFFFF$ .  
בתוצאת פעולת ה־ $and$ , שתיכתב באוגר  $t0$ , ערך סיבית יהיה 1 אם ורק אם ערך שני הערכים במקומות המתאימים יהיה 1. הדבר מתקיים ב־8 הסיביות המשמעותיות של שני המספרים ( $0xFF=1111\ 1111$ ) וכן בספרות ה־9 עד ה־16 מימין.  
סה"כ התוצאה תהיה  $0xFF00FF00$ .

איך תשתנה תשובתכם נחליף את הפקודה בשורה (4) ובמקום  $sra$  נבצע  $srl$ ?  $0x0000FF00$ .  
חישוב והסבר:

בשיפט לֹוגִי ימינה 16 מקומות, יתווספו משמאל 16 סיביות 0 למספר וייכנס ל־ $t2$  הערך  $0x0000FF00$ .  
אי לכך, תוצאת פעולת  $s$  and תהיה  $0x0000FF00$ .

2. הפסיאודו פקודה בשורה (2) מתרגמת לשתי פקודות אמיתיות  $lui\ ori$  ראו עמודת Basic. במידה והיינו מחליפים את פקודה ה־ $ori$  בפקודת  $addi$  האם תשובתכם לסעיף א תשתנה? תשתנה.  
חישוב והסבר:

פקודת ה־ $lui$  המיידית מעתיקה את 16 הביטים המידיים בה אל 16 המקומות המשמעותיים באוגר שנמצא באופרנד שלה ומוסיפה להם משמאל 16 אפסים. לכן, לאחר ביצוע הפקודה יהיה באוגר  $at$  הערך  $0xFF000000$ .

הוספה של הערך  $0x0000FFFF$ , היא למעשה הוספה של המספר  $0xFFFF$ , שלאחר ההרחבה שומרת-הסימן ייכנס ל־ALU כמספר  $0xFFFFFFFF$ . כמובן שהוספה של מספר זה תגרור תוצאה שונה מאשר אם היינו מוסיפים למספר את  $0xFFFF$  ומקבלים את המספר הרצוי לנו.

במקום האופרנד המיידִי  $0xFF00FFFF$ , היה מתקבל המספר  $0xFFEFFFFFFF$ , וכמובן תתקבל תוצאה שונה לאחר תוצאת ה־ $and$  - רק ביט יחיד בכל אחד מן האופרנדים יהיה 0, ובהתאם בדיוק שני ביטים יהיו 0 במספר השמור ב־ $t0$ , מספר השונה ממספר הביטים במספר השמור שם כעת.

3. מתוך תמונת סגמנט הקוד, מהי הכתובת של התווית **Panda**, הרשומה בשורה (5)?  
הכתובת  $0x00400058$ .  
חישוב והסבר:

לפי תמונת סגמנט הקוד, תתבצע במהלך ביצוע שורת ה־ $jump$  ההוראה  
$$PC \leftarrow PC + 4 + 4 \cdot 0x00000011 = 0x00400010 + 0x4 + 0x0000004$$
$$= 0x00400058$$

את שתי שאלות התכנות יש לכתוב בנפרד : כל אחת בקובץ asm/.s.  
נפרד. קראו היטב את כל ההוראות המופיעות בממ"ן. אי שמירה  
אפילו על חלק מההוראות הללו עלולה להוביל לפגיעה קשה בציון  
שתקבלו על הממ"ן.

את שלושת הקבצים – קובץ זה ושני קבצי asm/.s – יש לארוז בתוך  
קובץ zip ולהעלות רק אותו למערכת המטלות.